What is claimed is:

1. A processor comprising:

a front end that obtains instructions; and

a back end that provides speculative execution of the instructions;

wherein the processor has an instruction set architecture including speculative execution control circuitry that handles at least one machine instruction that facilitates synchronization between parallel processes by exposing the processor speculation to program control.

2. The processor of claim 1, wherein the front end comprises an in-order front end, and the back end comprises an out-of-order execution engine, which re-orders the instructions, and one or more execution units that perform the re-ordered instructions.

3. The processor of claim 2, wherein the out-of-order execution engine comprises an out-of-order execution management unit, including at least one buffer, and an in-order retire-store unit.

4. The processor of claim 3, wherein the at least one buffer comprises a reorder buffer.

5. The processor of claim 1, wherein the front end comprises a fetch-decode unit and a branch prediction unit.

6. The processor of claim 1, wherein the at least one machine instruction comprises:

a speculative execution instruction that takes first and second operands, behaves as a no-op if a memory location indicated by the first operand contains a first value, causes the processor to speculatively execute additional instructions if the memory location contains a second value, and causes the processor to start executing instructions from an address indicated by the second operand if a mis-speculation occurs; and

a speculation termination instruction that takes first and second operands and causes the processor to begin retiring the additional instructions if the additional instructions have been speculatively executed.

7. The processor of claim 6, wherein the mis-speculation comprises an interrupt.

8. A processor comprising:

a front end that obtains instructions; and

a back end that provides speculative execution of the instructions

wherein the processor has an instruction set architecture including speculative execution control circuitry that handles a speculative execution instruction and a speculation termination instruction.

9. The processor of claim 8, wherein the front end comprises an in-order front end, and the back end comprises an out-of-order execution engine, which re-orders the instructions, and one or more execution units that perform the re-ordered instructions.

10. The processor of claim 8, wherein the speculative execution instruction comprises an instruction that takes first and second operands, causes the processor to speculatively execute additional instructions if a memory location contains a value, and causes the processor to start executing instructions from an address indicated by the second operand if a mis-speculation occurs, and the speculation termination instruction comprises an instruction that causes the processor to begin retiring the additional instructions.

11. A machine-implemented method comprising:

generating parallel processes in a data processing machine;

effecting synchronization between the parallel processes using processor speculation in the data processing machine; and

providing output resulting from the synchronized parallel processes.

12. The method of claim 11, wherein said generating parallel processes comprises running a software program that spawns multiple threads in the data processing machine.

13. The method of claim 11, wherein said effecting synchronization comprises translating at least one high-level software instruction into at least one machine instruction that controls speculative execution in a processor.

14. The method of claim 11, wherein said effecting synchronization comprises placing in an out-of-order execution management unit of a processor, at least one machine instruction that limits when other machine instructions are retired from the out-of-order execution management unit.

15. The method of claim 11, wherein said providing output comprises sending the output to another data processing machine.

16. An article comprising a machine-readable medium embodying information indicative of instructions that when performed by one or more machines result in operations comprising:

generating parallel processes in a data processing machine;

effecting synchronization between the parallel processes using processor speculation in the data processing machine; and

providing output resulting from the synchronized parallel processes.

17. The article of claim 16, wherein said generating parallel processes comprises running a software program that spawns multiple threads in the data processing machine.

18. The article of claim 16, wherein said effecting synchronization comprises translating at least one high-level software instruction into at least one machine instruction that controls speculative execution in a processor.

19. The article of claim 16, wherein said effecting synchronization comprises placing in an out-of-order execution management unit of a processor, at least one machine instruction that limits when other machine instructions are retired from the out-of-order execution management unit.

20. The article of claim 16, wherein said providing output comprises sending the output to another data processing machine.

21. A machine-implemented method comprising:

speculatively executing machine instructions, including a memory access instruction, in a processing system to effect synchronization between parallel processes;

retiring the speculatively executed machine instructions; and

maintaining cache coherence in the processing system during said executing and said retiring to identify a mis-speculation to effect the synchronization between the parallel processes.

22. The method of claim 21, wherein said maintaining cache coherence comprises providing invalidation based cache coherence.

23. The method of claim 21, wherein said speculatively executing machine instructions comprises speculatively executing machine instructions in the processing system comprising multiple processors, and the mis-speculation comprises a memory dependency violation.

24. The method of claim 21, wherein the mis-speculation comprises at least one of an interrupt, an external event, and a memory dependency violation.

25. A system comprising:

a processor having a processor architecture that provides speculative execution of machine instructions and exposes said speculative execution to program control through at least one machine instruction; and

a memory coupled with the processor, the memory embodying information indicative of instructions, including the at least one machine instruction, that result in synchronization between parallel processes when performed by the processor with detection of mis-speculation.

26. The system of claim 25, wherein the processor comprises a uniprocessor.

27. The system of claim 25, wherein the processor comprises a multiprocessor.

28. The system of claim 27, wherein the multiprocessor comprises multiple processing units on a single die.

29. The system of claim 25, further comprising:

a communication interface; and

a virtual machine that translates the information, received via the communication interface, into the at least one machine instruction.

30. The system of claim 29, wherein the virtual machine comprises a Java virtual machine.

31. The system of claim 25, wherein the at least one machine instruction comprises:

a speculative execution instruction that takes first and second operands, behaves as a no-op if a memory location indicated by the first operand contains a first value, causes the processor to speculatively execute additional instructions if the memory location contains a second value, and causes the processor to start executing instructions from an address indicated by the second operand if a mis-speculation occurs; and

a speculation termination instruction that causes the processor to begin retiring the additional instructions if the additional instructions have been speculatively executed.

32. The system of claim 25, further comprising an environmental sensor coupled with the processor.

33. A processing system comprising:

processing means for speculatively executing machine instructions in response to a speculative execution instruction, including means for detecting a mis-speculation;

means for treating multiple speculative instructions as a group for purposes of retirement such that the multiple speculative instructions are flushed from the processing means together and execution proceeds from an address in response to a detected mis-speculation to effect synchronization between parallel processes.

34. The processing system of claim 33, wherein said means for detecting a mis-speculation comprises means for maintaining cache coherence in the processing means.